MANAV RACHNA INTERNATIONAL UNIVERSITY

Research Plan
For
Ph. D. Programme 2009-10

TITLE: FPGA Based Low Power Optimal Digital System Design

DEPARTMENT OF ELECTRONICS & COMMUNICATION

FACULTY OF ENGINEERING & TECHNOLOGY

Submitted by:

Name: Naresh Grover
Signature:
Registration No. : 09019990011
Contact No. 9910163137
Email ID: grovernr@rediffmail.com

Supervisor
Name: Dr. M. K. Soni
Signature:
Designation: Exec. Dir & Dean, FET, MRIU
Contact No.: 09871588132
Email ID: ed.fet@mriu.edu.in

Co-Supervisor
Not Applicable

ABSTRACT

Due to the flexibility, programmability and low end product life cycle, Field Programmable Gate Arrays FPGAs are highly desirable for implementation of digital systems. Since the introduction of FPGA, research and development has produced dramatic improvements in its speed and area efficiency, narrowed the gap between FPGAs and ASICs and made FPGAs the platform of choice for implementing digital circuits. Unfortunately, the advantages of FPGAs due to their high power consumption and area are still offset in many cases. The territories of FPGAs applications can be expanded more effectively by reducing the power consumption without sacrificing much performance or incurring a large chip area. Various techniques are used at system, device, and circuit and architecture level for reduction of power consumption of FPGAs and every technique has its merits and limitations. We propose to design and implement the digital system on FPGA and then to optimize it for minimum power consumption without affecting its performance by interfacing Xilinx with MAT lab using Simulink HDL coder and using available optimization tool that shows an improvement over the earlier similar techniques of reducing the power consumption at design level of FPGA based system.

Key Words- Static and dynamic power, embedded memories, clock gating, glitches, logic power, Modelsim Simulator.

CONTENTS

S. No.	Description	Page No.
1	Introduction	1-3
2	Literature Review	3-13
3	Description of Broad Area/Topic	13-14
4.	Problem Identification/Objective of the Study	14
5	Methodology to be adopted	15
6	Proposed/expected outcome of the research	15
7	References	15-19

1. Introduction

Field-Programmable Gate Arrays (FPGAs) are integrated circuits (ICs) that can be programmed to implement any digital circuit. The main difference between FPGAs and conventional fixed logic implementations, such as Application Specific Integrated Circuits (ASICs), is that the designer can program the FPGA on-site, non-recurring engineering (NRE) costs gets eliminated and it reduces time-to-market significantly. FPGAs are approximately 3 times slower, 20 times larger, and 12 times less power efficient compared to ASICs [1] because its programmable switches controlled by configuration memory occupy a large area and add a significant amount of parasitic capacitance and resistance to the logic and routing resources.

There has been focus on faster and more area efficient programmable routing resources in by the researchers. The Versatile Place and Route (VPR) tool described in [2], yields significant improvements in performance by improving on the existing clustering, placement, and routing algorithms. Logic-to-memory mapping tools, described in [3]-[5], shows improvement in the area efficiency of FPGAs with embedded memories wherein parts of the application are packed into unused memories before mapping the rest of the application into logic elements.

In recent years, the main emphasis of the research has been shifting to lower the power consumption which is an important part of equation determining the product size, weight and efficiency. The advantages of FPGAs high power consumption and area are the offset in many cases. The ever-growing demand for low-power portable communications and computer systems is motivating new low power techniques, especially for FPGAs, which dissipate significantly more power than fixed-logic implementations. Indeed, the International Technology Roadmap for Semiconductors (ITRS) has identified low-power design techniques as a critical technology need [5].

Like all integrated circuits, FPGAs also dissipate two types of power i.e. static and dynamic power. Static power is consumed due to transistor leakage and is dissipated when current leaks from the power supply to ground through transistors that are in the “off-state” due to three types of leakages : sub-threshold leakage (from source to drain), gate-induced drain leakage, and gate direct-tunneling leakage. Dynamic power is consumed mainly by toggling nodes as a function of

voltage, frequency, and capacitance and is dissipated when capacitances are charged and discharged during the operation of the circuit and consumed during switching events in the core or I/O of FPGA. The dynamic power consumption is generally modeled as below:

$$P = \sum_i C_i . V_i^2 . f_i$$

where C,V and f represent capacitance, the voltage swing, and clock frequency of the resource i, respectively [6]. The total dynamic power consumed by a device is the summation of the dynamic power of each resource. Because of programmability of FPGA the dynamic power is design-dependent and the factors that contribute to the dynamic power are: the effective capacitance of resources, the resources utilization, and the switching activity of resources [6],[7]. The effective capacitance corresponds to the sum of parasitic effects due to interconnection wires and transistors. Since FPGA architecture usually provides more resources than required to implement a particular design, some resources are not used after chip configuration and they do not consume the dynamic power (this is referred to as resource utilization). Switching activity represents the average number of signal transitions in a clock cycle. Though generally it depends on the clock itself, it may also depend on other factors (e.g. temporal patterns of input signals). Hence, the above equation can be rewritten as:

$$P = V^2 . f . \sum_i C_i . U_i . S_i$$

where V is the supply voltage, f is the clock frequency, and C , U , and S , are the effective capacitance, the utilization, and the switching activity of each resource, respectively.

FPGAs consume much more power than its counterpart ASICs because they have a large number of transistors per logic function in order to program the device. FPGA contains a large number of configuration bits, both within each logic element and in the programmable routing used to connect logic elements. This extra circuitry provides flexibility but it affects both the static and dynamic power dissipated by the FPGA.

The power consumption in FPGA can be reduced at four levels: chip/architecture, system, designer and operation level. According to authors in [8], there are three major strategies in

FPGA power consumption reduction. First, changes can be done at the system level (e.g. simplification of the algorithms used). Secondly, if the architecture of FPGA is already fixed, a designer may change the logic partitioning, mapping, placement and routing. Finally, if no changes at all are possible, enhancing operating conditions of the device may be still promising (this includes changes in the capacitance, the supply voltage, and the clock frequency).

For the FPGA based circuits, 80% of the total power is consumed at the level of system design and RTL and rest 20% of the power is at Gate and transistor level. In the research work we intend to design and implement the digital system on FPGA and then to optimize it for minimum power consumption without affecting its performance by interfacing Xilinx with MAT lab using Simulink HDL coder and using available optimization tool that shows an improvement over the earlier similar techniques of reducing the power consumption at design level of FPGA based system.

Section 2 contains the detail of already published work for reducing the power consumption in FPGA based circuits, section 3 contain the description of research topic, section 4 contains the objective of problem identification, section 5 contains the division of tasks into subtasks and proposed methodologies for individual subtasks, section 6 contains the expected outcome of the research and section 7 contains references.

2. Literature Review

The work carried out and various design techniques used for power reduction in FPGAs are reviewed and discussed in this section.

Tuan and Lai in [9] [32] examined leakage in the Xilinx Spartan-3 FPGA, a 90nm commercial FPGA. Table 1 shows the breakdown of leakage in a Spartan-3 CLB, which is similar to the Virtex-4 CLB. Leakage is more in the interconnects, configuration SRAM cells, and to a lesser extent, LUTs. Combined, these structures account for 88% of total leakage.

Source of Power	Percentage of Total Leakage Power
LUTs	34
Interconnects	16
Configuration SRAMs	38
Others	12

Table 1 Breakdown of leakage in a Spartan-3 CLB

As pointed out by Taun and Lai [9], the contents of an FPGA's configuration SRAM cells change only during the FPGA's configuration phase. Configuration is normally done once at power-up. Therefore, the speed performance of an FPGA's SRAM configuration cells is not critical, as it does not affect the operating speed of the circuit implemented in the FPGA. The SRAM cells can be slowed down and their leakage can be reduced or eliminated using previously-published low leakage memory techniques or by implementing the memory cells with high-VTH or long channel transistors. Leakage was not a primary consideration in the design of Spartan-3. If SRAM configuration leakage were reduced to zero, the Spartan-3 interconnects and LUTs would account for 55% and 26% of total leakage, respectively.

A number of recent papers have considered the breakdown of dynamic power consumption in FPGAs [10, 11, 12]. In [12] Shan studied the breakdown of power consumption in the Xilinx Virtex-II commercial FPGA. The results are summarized in Table 2. Interconnect, logic, clocking, and the I/Os were found to account for 60%, 16%, 14%, and 10% of Virtex-II dynamic power, respectively. A similar breakdown was observed by Poon, Yan and Wilton in [10]. As per Yeap [13], the FPGA power breakdown differs from that of custom ASICs, in which the clock network is often a major source of power dissipation.

Source of Power	Percentage of Total Dynamic Power
Interconnects	60
Logic	16
Clocking	14
I/O	10

Table 2: Breakdown of power consumption in the Xilinx Virtex-II

It is understood that the dominance of interconnect in FPGA dynamic power is mainly due to the composition of FPGA interconnect structures that consist of pre-fabricated wire segments, with used and unused switches attached to each wire segment. Furthermore, SRAM configuration cells and circuitry constitute a considerable fraction of an FPGA's total area. For example, Ye, Rose and Lewis [14] suggest that more than 40% of an FPGA's logic block area is SRAM configuration cells. Interconnect thus presents a high capacitive load in FPGAs, making it the primary source of dynamic power dissipation.

Vendors such as Altera and Xilinx in their latest FPGA devices, incorporate various low-power device-level technologies. Traditional FPGAs and ASICs used only two oxide thicknesses (dual oxide): a thin oxide for core transistors and a thick oxide for I/O transistors. Moving toward high-performance 90 nm FPGAs, Xilinx integrated circuit (IC) designers started to adopt the use of a third-gate oxide thickness (triple oxide) of midox in the transistors of the 90 nm Virtex™-4 FPGAs that allows a substantial reduction in overall leakage and static power, compared to other competitive FPGAs. Subsequent versions of Virtex-5 FPGAs and above continue to deploy the triple oxide technology in the 65 nm process nodes to enable a significant lower leakage current of about 38% lower than that for a 65 nm device. At the device level, Altera and Xilinx both utilize triple gate oxide technology, which provides a choice of three different gate thicknesses, to trade-off between performance and static power [15, 16].

Calhoun proposes the creation of fine-grained “sleep regions", making it possible for a logic block's unused LUTs and flip-flops to be put to sleep independently [17]. A more coarse-grained sleep strategy was proposed in [18], which partitioned an FPGA into entire regions of logic

blocks, such that each region can be put to sleep independently. The authors restricted the placement of the implemented design to fall within a minimal number of the pre-specified regions, and studied the effect of the placement restrictions on design performance.

Rahman addressed leakage in FPGA interconnects [19] that applied well-known leakage reduction techniques to interconnect multiplexers and proposed four different techniques. First, extra configuration SRAM cells were introduced to allow for multiple OFF transistors on unselected multiplexer paths. The intent is to take advantage of the "stack effect". A second approach described by the author is to layout portion of the multiplexer in separate wells, allowing body-bias techniques to be used to raise the VTH of multiplexer transistors that are not part of the selected signal path. As a third technique, they proposed negatively biasing the gate terminals of OFF multiplexer transistors. The negative gate bias leads to a significant drop in sub threshold leakage. Finally, the authors proposed using dual-VTH techniques, wherein a subset of multiplexer transistors are assigned high-VTH (slow/low leakage), and the remainder of transistors are assigned low-VTH (fast/leaky). The dual-VTH idea, impacts FPGA router complexity, as the router must assign delay-critical signals to low-VTH multiplexer paths. A more recent paper by Ciccarelli applies dual-VTH techniques to the routing switch buffers in addition to the multiplexers [20].

FPGA vendors have recently started to increase the size of embedded memory cells in their FPGAs. The trend can be noticed toward increasing the ratio of memory bits to logic cells in low-cost FPGAs. It can further be deduced that minimizing leakage power in the FPGA embedded memory is more beneficial than minimizing that of the logic cells. In [21] Meng and others proposed a CAD technique to reduce leakage power dissipation in FPGA embedded memory bits by adding path traversal and location assignment techniques in the embedded memory mapping. The authors assume that all the embedded memory cells can support the drowsy mode by having the ability to connect to two supply voltages VDDH and VDDL, a high and low supply voltage respectively. The cell still retains the stored data even while the memory bit is operating at the low supply voltage but the bit will consume less leakage power as leakage power is proportional to the supply voltage. This scheme is referred to as drowsy memory for memory bits.

Several techniques to reduce leakage power in memories using the drowsy scheme have been proposed in the literature; however, they mainly targeted cache memories in processors [22]. FPGA embedded memories access are statically scheduled and the data is stored statically whereas cache memories have variable latencies and dynamic data placement have different characteristics. Hence, the problem of minimizing leakage power in FPGA embedded memories helps in finding the best static layout of the data in the memory to have maximum leakage savings from putting the memories into a low-leakage state. The unused memory entries are placed in a sleep mode and are not to be brought out of that mode, while the used memory entries are put in a sleep mode when they are not accessed and are waken up when needed.

In [21] Meng and others proposed three different modes: sleep mode, drowsy mode, and live mode. The sleep mode is used for unused memory entries by shutting down the supply voltage from the unused memory bits. In the study the authors showed that just by putting the unused memory entries in the sleep mode (used-active), one can save an average of 36% of the memory leakage power without utilizing any scheme for dynamically waking up(or putting to sleep) the used memory entries. Moreover, in the embedded memories, on an average about 75% of leakage power savings can be achieved just by using the minimum number of memory entries and turning off the unused entries (min-entry). It is noticed that the drowsy-long scheme offers an additional 10% leakage power savings over the min-entry scheme [21]. Moreover, the path-place algorithm on an average achieves about 95% leakage power savings. Hence, it can be concluded that the two best memory layout techniques are the min-entry and path place techniques. The min-entry scheme offers very good leakage power savings in terms of both computational time and extra circuitry needed by the FPGA since it only supports active and off modes. On the other hand, the path-place scheme supports three memory modes: active, low leakage with data retention, and off modes [21].

Kumar and Anis [23] proposed two architectures i.e. a homogeneous and a heterogeneous architectures. The homogeneous architecture uses inside the cluster sub blocks of different VTH, while the heterogeneous architecture uses interleaved two types of clusters, where one of the clusters is composed of low VTH logic cells and the other consists of low and high VTH logic cells. The authors proposed a CAD framework that starts by assigning the whole design to high

VTH logic cells. Then the algorithm starts assigning the logic cells into low VTH cells as long as the cell has positive slack and the new path slack does not become negative. The algorithm clusters the logic cells into the clusters that correspond to the architecture being used in the next stage. Finally, constrained placement is used to place the clustered designs into the FPGA architecture. It was noticed that both the homogeneous and heterogeneous architectures result in very close leakage power savings with almost equal delay penalties.

Lewis and others proposed the use of body biasing in FPGAs to slow down the cells on non critical paths to achieve a reduction in the sub threshold leakage power [24]. The authors concluded that using a granularity that is equal to two clusters results in considerably sufficient amount of leakage power savings without incurring big penalties on both the delay and area of the FPGA.

The first comprehensive effort to develop a low-energy FPGA was by a group of researchers at UC Berkeley [25, 26, 27]. Power reductions were achieved through following significant changes in the logic and routing fabrics:

- Larger, 5-input LUTs were used rather than 4-LUTs, allowing more connections to be captured within LUTs instead of being routed through the power- dominant interconnect.
- A new routing architecture was deployed, combining ideas from a 2-dimensional mesh, nearest-neighbor interconnects, and an inverse clustering scheme.
- Specialized transmitter and receiver circuitry were incorporated into each logic block, allowing low-swing signaling to be used.
- Double-edge-triggered flip-flops were used in the logic blocks, allowing the clock frequency to be halved, and reducing clock power.

The main limitations of the work are:

- The proposed architecture represents a “point solution" in that the effect of the architectural changes on the area-efficiency, performance, and routability of real circuits was not considered
- The basis of the architecture is the Xilinx XC4000, which was introduced in the late 1980s and differs considerably from current FPGAs
- The focus was primarily on dynamic power and leakage was not a major consideration.

Li, Chen, He and Cong considered power trade-offs at the architectural level [11] , which examined the effect of routing architecture, LUT size, and cluster size (the number of LUTs in a logic block) on FPGA power-efficiency. Using the metric of power-delay product, authors suggest that 4-input LUTs are the most power-efficient, and that logic blocks should contain twelve 4-LUTs. In these studies, despite their focus on power, power-aware CAD tools were not used in the architectural evaluation experiments. The architectures evaluated in the UC Berkeley work [26] are somewhat out-of-step with current commercial FPGAs. The authors in [11] suggest that a mix of buffered and un-buffered bidirectional routing switches should be used but the modern commercial FPGAs no longer use un-buffered routing switches; rather, they employ unidirectional buffered switches.

Li and others applied the dual-VDD concept to FPGAs [28] and proposed heterogeneous architecture in which some logic blocks are fixed to operate at high-VDD (high speed) and some are fixed to operate at low-VDD (low-power, but slower). The power benefits of the heterogeneous fabric were found to be minimal mainly due to the rigidity of the fixed fabric and the performance penalty associated with mandatory use of low-VDD in certain cases.
Subsequently the same authors extended their dual-VDD FPGA work to allow logic blocks to operate at either high or low-VDD [29] and by using such “configurable" dual-VDD schemes, power reductions of 9-14% (versus single-VDD FPGAs) were reported. A limitation of [28] and [29] is that the dual-VDD concepts were applied only to logic and not to interconnect, where most power is consumed and was assumed to always operate at high-VDD.

Gayasen and others overcame this limitation which apply dual-VDD to both logic and interconnect [18]. A dual-VDD FPGA presents a more complex problem to FPGA CAD tools. CAD tools need to select specific LUTs to operate at each supply voltage, and then assign these LUTs to logic blocks with the appropriate supply. Chen and others developed algorithms for dual- VDD mapping and clustering to address these issues in conjunction with the architecture work mentioned above [30, 31].

According to Lee, Nam and Chang [8], there are following three major strategies in FPGA power consumption reduction:

- First, changes can be done at the system level (e.g. simplification of the algorithms used).
- Secondly, if the architecture of FPGA is already fixed, a designer may change the logic partitioning, mapping, placement and routing and
- Finally, if no changes at all are possible, enhancing operating conditions of the device may be still promising (this includes changes in the capacitance, the supply voltage, and the clock frequency).

Following basic techniques have been explored so far at system level design:
Kuon and Rose suggested to use coarse-grained embedded blocks rather than the fine-grained configurable logic blocks in an FPGA, since the former are more power efficient than the latter for the same function. [32]. For using course-grained FPGAs, it is to be ensured that power consumption for routing would not increase significantly.

Osborne and others used clock gating as a simple and effective method for reducing dynamic power consumption. It reduces the dynamic power by eliminating unnecessary toggling on the outputs of flip-flops of a circuit, gates in the fan-out of the flip-flops, and clock signals. Clock gating can be used to reduce dynamic power consumption to prevent signal transitions by disabling the clock for the inactive regions. The circuitry in an operator is gated when not in use if it can be combined with word-length optimization [33].

Wilton, Ang and Luk found that, at a given clock speed, pipelining which is a simple and effective way of reducing glitching can reduce the amount of energy per operation by between 40% and 90% for applications such as integer multiplication, CORDIC, triple DES, and FIR filters [34].

Chow et al observed that power reduction between 4% and 54% can be achieved for various arithmetic circuits [35] by using dynamic voltage scaling to adapt the dynamic supply voltage to the FPGA as the temperature changes.

A number of low-power techniques have also been incorporated into the commercial FPGA CAD tools. Detailed power models have been integrated within the Altera Quartus II [36] and Xilinx ISE CAD tools [37] that provide a spreadsheet utility to make early power predictions

before the design is complete and a detailed power model that can be used when the design is complete.

Tessier described that power is also minimized by optimizing the mapping to the embedded memories and to the embedded DSP blocks [38]. In ISE, power is minimized during placement and routing by minimizing the capacitance of high-activity signals. Dynamic power dissipation is further minimized by strategically setting the configuration bits within partially used LUTs to minimize switching activity.

A number of studies have investigated low-power FPGA architecture design.
George and others have described energy-efficient FPGA routing architectures and low-swing signaling techniques to reduce power. [39].
Sivaswamy and others proposed a new FPGA routing architecture that utilizes a mixture of hardwired and traditional programmable switches [40] that reduces static and dynamic power by reducing the number of configurable routing elements. As the architecture and the circuit-level implementation of the FPGA directly affects the efficiency of mapping applications to FPGA resources and the amount of circuitry to implement these resources, these implementations are the main keys in reducing power.

Kusse and Rabaey introduced the energy-efficient modules for embedded components in FPGAs to reduce power by optimizing the number of connections between the module and the routing resources, and by using reduced supply voltage circuit techniques [41]. Anderson and Najm [42] presented a novel FPGA routing switch with high-speed, low-power, or sleep modes that reduces dynamic power for non timing critical logic and standby power for logic when it is not being used.

Lin and others applied power-gating to the switches in the routing resources to reduce static power and duplicate routing resources that use either high or low Vdd [43].
A recent study in [44] suggests that glitching accounts for 31% of dynamic power dissipation in FPGAs. Glitching occurs when values at the inputs of a LUT toggle at different times due to uneven propagation delays of those signals. Lamoureux and others propose a method for minimizing glitching by adding configurable delay elements to the inputs to each logic element

in the FPGA. On an average, the proposed technique eliminates 87% of the glitching that reduces overall FPGA power by 17% at the cost of the overall FPGA area by 6% and critical-path delay by less than 1% due to the added circuitry increases.

Glitch reduction techniques can be applied at various stages in the CAD flow. Since glitches are caused by unbalanced path delays to LUT inputs, it is natural to design algorithms that attempt to balance the delays.

Cheng and others proposed that reduction of glitches can be done at the technology mapping stage, in which the mapping is chosen based on glitch-aware switching activities [45]. Czajkowski and Brown [46] proposed another approach that operates at the routing stage, in which the faster arriving inputs to a LUT are delayed by extending their path through the routing network and delay balancing can also be done at the architectural level. However, these approaches all incur an area or performance cost.

Shum and Anderson in their paper present a glitch reduction optimization algorithm based on don't-cares that sets the output values for the don't-cares of logic functions in such a way that reduces the amount of glitching [47]. The authors performed the process *after* placement and routing, using timing simulation data to guide the algorithm.

Wang and others observed that the dynamic power consumption is supposed to increase linearly with changes of clock frequency and size of a design. It was observed that with the clock frequency decrease the effect of the design size on power consumption gets decreased [48]. The authors mentioned that FPGA designs can be enlarged with a disproportionally low dynamic power increase as long as the device operates at low frequencies. Only at the highest frequencies, the dynamic power changes proportionally to the design area.

A lot of design performance like FPGA area utilization and power consumption get affected by coding style using HDL. Dollas and others reported very interesting results by demonstrating how an HDL behavioral approach leads to more efficient implementations comparing to

structural descriptions [49]. Garrault and Phiofsky also suggest to describe designs behaviorally as much as possible [50].

Although many of the significant improvements have been made to improve power and energy efficiency of FPGAs, ranging from low level processes and circuit design techniques to high level techniques, but many opportunities to further reduce power in FPGAs remain. While further improvements will likely be made at all levels, there seems to be significant potential for power savings at the system level, where power reduction can be obtained by optimizing management and scheduling of system resources.

3. Description of Broad Area

The process of digital hardware has changed dramatically over past few years with the development of field programmable devices (FPDs). Now FPGAs are attractive architecture choice for any digital system design due to its flexibility, reprogrammability and high performance. FPGAs are replacing ASICs in many applications because the price of a single chip development shoots very high in each successive technology iteration, whereas the relative prices of FPGA architectures are becoming more and more attractive. Many other advantages of FPGAs such prototyping and its field reprogrammability make the more viable alternative in many current ASIC applications. These advantages of FPGAs are offset by its high power consumption and large area in many cases. Kuon and Rose have shown that FPGAs can be up to 40 times larger and consume 12 times more dynamic power than its equivalent ASIC implementation [51]

Due to the rapid growth of battery powered devices and portable digital applications; the more attention has been attracted towards power minimization and its optimization. There is a constraining factor of low power consumption for FPGA to enter main stream low power applications.

The proposed broad area of my research would be FPGA based optimal design of a comprehensive digital system after studying the various FPGA based low power design techniques. From the study it is clear that almost 80% of the total power is at system and RTL level whereas balance 20% is at gate and transistors level. Our focus of research will be on

designing any comprehensive FPGA based digital system and then to explore the optimization techniques to reduce its power to have an optimal design of that digital system.
To get the maximum performance of from any FPGA based design, the coding of RTL description should be proper.

It is proposed to write VHDL code for a comprehensive 32 bit floating point Arithmetic Unit that performs all four arithmetic operations i.e. addition, subtraction, multiplication and division. The VHDL design code would then be implemented on FPGA platform and the parameters would be analysed mainly in respect of area and power. The VHDL code so implemented would then be converted on to MAT lab platform using Simulink model for verification of VHDL code in Modelsim, where the same shall be optimized on power using optimization techniques and the optimized code so obtained shall again be converted back to FPGA platform, implemented and analysed.

4. Objective of Study

FPGAs are gaining more and more popularity in the digital system design over general purpose processors and application specific integrated circuits because of its flexibility, field programmability and re-configurability. However due to its high power consumption as compared to its counterparts, its applications are limited in the digital systems of low power applications. The main aim is to study the various low power design techniques used for FPGA based digital systems. The focus will be to design a comprehensive digital system by writing VHDL code and implementing the same on Xilinx FPGA platform and then to explore the optimization techniques to have low power optimal design of the same designed digital system.
The main objective of the research would be:

- To study various low power design techniques used for FPGA based digital system
- To analyse FPGA performance
- To design a FPGA based low power optimal comprehensive digital system

5. Methodology to be adopted

The following steps and methodology shall be adopted to complete the proposed research work:

- Study the different FPGA architectures and its comparison
- Literature survey and study the related work done in respect of FPGA based low power digital system designs
- Study the Finite State Machine (FSM) design methods
- Study VHDL design methods for FSM
- Study FSM design on FPGAs
- Study Xilinx EDA tools
- Designing, implementation and analysis of 32 bit floating point arithmetic unit on Xilinx FPGA platform
- Study MAT Lab Tools Modelsim simulator
- Create Simulink model for verification of VHDL code in Modelsim
- Study and apply optimization techniques on MAT lab
- Create VHDL code in Modelsim for optimized simulink model
- Implement optimized VHDL code and analyse on Xilinx platform

6. Expected outcome of research

Every technique used at system, device, and circuit and architecture level for reduction of power consumption of FPGAs has its merits and limitations. Proposed research work is expected with an outcome of a FPGA based low power optimal design of a 32 bit floating point arithmetic unit for all four operations i.e. addition, subtraction, multiplication and division of two 32 bit floating point numbers, with minimum optimum power consumption without sacrificing the other parameters of the design. The method proposed in the research shall be implementable for any FPGA based digital system design with optimum power consumption and shall meet the objective of research.

References

[1] I. Kuon and J. Rose, "Measuring the Gap Between FPGAs and ASICs," ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp. 21-30, 2006.

[2] V. Betz., J. Rose, and A. Marquardt, “Architecture and CAD for deep-submicron FPGAs,” Kluwer Academic Publishers, 1999.

[3] J. Cong and S. Xu, Technology mapping for FPGAs with embedded memory blocks, in ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA), pp.179-188, 1998

[4] S. J. E. Wilton, “SMAP: heterogeneous technology mapping for area reduction in FPGAs with embedded memory arrays,” in ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA), pp. 171-178, 1998.

[5] International Technology Roadmap for Semiconductors, “International technology roadmap for semiconductors 2005,” 2005.

[6] L. Shang, A. S. Kaviani, and K. Bathala, “Dynamic Power Consumption in Virtex-II FPGA Family”, proceedings of the 2002 ACM/SIGDA 10th International Symposium on Field-Programmable Gate Arrays, pages157 – 164. ACM Press, 2002

[7] V. Degalahal and T. Tuan, “Methodology for high level estimation of FPGA power consumption”, Design Automation Conference, 2005. Proceedings of the ASP-DAC 2005. Asia and South Pacific, Volume 1, 18-21 Jan. 2005 Page(s):657 – 660 Vol. 1.

[8] H. G. Lee, S. Nam, and N. Chang, “Cycle-accurate energy measurement and high-level energy characterization of FPGAs”, Quality Electronic Design, 2003. Proceedings. Fourth International Symposium on 24-26 March 2003, Page(s):267 – 272

[9] T. Tuan and B. Lai. “Leakage Power Analysis of a 90nm FPGA". In: IEEE Custom Integrated Circuits Conference, pp. 57-60, San Jose, CA, 2003.

[10] K. Poon, A. Yan, and S. J. E. Wilton. “A Flexible Power Model for FPGAs". In: International Conference on Field-Programmable Logic and Applications, pp. 312-321, Montpellier, France, 2002.

[11] F. Li, D. Chen, L. He, and J. Cong. “Architecture Evaluation for Power-Efficient FPGAs". In: ACM/SIGDA International Symposium on Field Programmable Gate Arrays, pp. 175-184, Monterey, CA, 2003.

[12] L. Shang, A. Kaviani, and K. Bathala. Dynamic Power Consumption in the Virtex-II FPGA Family". In: ACM/SIGDA International Symposium on Field Programmable Gate Arrays, pp. 157{164, Monterey, CA, 2002.

[13] G. Yeap. Practical Low Power Digital VLSI Design. Kluwer Academic Publishers, Boston,

MA, 1998.

[14] A. Ye, J. Rose, and D. Lewis. “Using Multi-Bit Logic Blocks and Automated Packing to Improve Field-Programmable Gate Array Density for Implementing Datapath Circuits". In: IEEE International Conference on Field-Programmable Technology, pp. 129-136, Brisbane, Australia, 2004.

[15] Altera, “Quartus II Handbook,” Vol. 3, Chapter 10, 2007

[16] Xilinx, “Power Consumption in 65nm FPGAs,” 2007.

[17] B. Calhoun, F. Honore, and A. Chandrakasan. “Design Methodology for Fine-Grained Leakage Control in MTCMOS". In: ACM/IEEE International Symposium on Low-Power Electronics and Design, pp. 104-109, Seoul, Korea, 2003.

[18] A. Gayasen, Y. Tsai, N. Vijaykrishnan, M. Kandemir, M. Irwin, and T. Tuan. “Reducing Leakage Energy in FPGAs Using Region-Constrained Placement". In: ACM/SIGDA International Symposium on Field Programmable Gate Arrays, pp. 51-58, Monterey, CA, 2004.

[19] A. Rahman and V. Polavarapuv. “Evaluation of Low-Leakage Design Techniques for Field-Programmable Gate Arrays". In: ACM/SIGDA International Symposium on Field Programmable Gate Arrays, pp. 23-30, Monterey, CA, 2004.

[20] L. Ciccarelli, A. Lodi, and R. Canegallo. “Low Leakage Circuit Design for FPGAs". In: IEEE Custom Integrated Circuits Conference, pp. 715-718, Orlando, FL, 2004.

[21] Meng Y, Sherwood T, Kastner R. Leakage power reduction of embedded memories on FPGAs through location assignment. In: Proceedings of Design Automation Conference (DAC). 2006:612–617

[22] Kim NS, Flautner K, Blaauw D, et al. Circuit and micro architectural techniques for reducing cache leakage power. IEEE Trans Very Large Scale Integration (VLSI) Syst. 2004;12: 167–184.

[23] Kumar A, Anis M. Dual-threshold CAD framework for sub threshold leakage power aware FPGAs. IEEE Trans Computer Aided Des Integrated Circ Syst. 2007;26:53–66.

[24] Lewis D, Ahmed E, Cashman D, et al. Architectural enhancements in Stratix-IIITM and Stratix IVTM. In: Proceedings of FPGA. 2009:33–41

[25] E. Kusse and J. Rabaey. “Low-Energy Embedded FPGA Structures". In: ACM/IEEE International Symposium on Low-Power Electronics Design, pp. 155-160, Monterey, CA, 1998.

[26] V. George, H. Zhang, and J. Rabaey. “The Design of a Low Energy FPGA". In: ACM

International Symposium on Low Power Electronics and Design, pp. 188-193, San Diego, CA, 1999.

[27] V. George and J. Rabaey. Low-Energy FPGAs: Architecture and Design. Kluwer Academic Publishers, Boston, MA, 2001.

[28] F. Li, Y. Lin, L. He, and J. Cong. “Low-Power FPGA Using Pre-Defined Dual-Vdd/Dual-Vt Fabrics". In: ACM/SIGDA International Symposium on Field Programmable Gate Arrays, pp. 42-50, Monterey, CA, 2004.

[29] F. Li, Y. Lin, and L. He. “FPGA Power Reduction Using Configurable Dual-Vdd". In: ACM/IEEE Design Automation Conference, pp. 735-740, San Diego, CA, 2004.

[30] D. Chen, J. Cong, F. Li, and L. He. “Low-Power Technology Mapping for FPGA Architectures with Dual Supply Voltages". In: ACM/SIGDA International Symposium on Field Programmable Gate Arrays, pp. 109-117, Monterey, CA, 2004

[31] D. Chen and J. Cong. “Register Binding and Port Assignment for Multiplexer Optimization". In: IEEE/ACM Asia and South Pacific Design Automation Conference, pp. 68-73, Yokohama, Japan, 2004.

[32] I. Kuon and J. Rose, “Measuring the gap between FPGAs and ASICs,” IEEE Trans. on Computer-Aided Design, vol. 26, no. 2, pp. 203-215, Feb. 2007.

[33] W.G. Osborne, W. Luk, J.G.F. Coutinho and O. Mencer, “Power and branch aware word-length optimization,” Proc. IEEE Symp. on Field-Prog. Custom Computing Machines, IEEE Computer Society Press, 2008.

[34] S.J.E. Wilton, S-S. Ang, and W. Luk. “The impact of pipelining on energy per operation in field programmable gate arrays”. In Proc. Field Prog. Logic and Applications, LNCS 3203, pp. 719–728, 2004.

[35] C.T. Chow et al, “Dynamic voltage scaling for commercial FPGAs,” Proc. IEEE Int. Conf. on Field Prog. Technology, 2005.

[36] Altera, “Quartus II Handbook,” Vol. 2, Chapter 9, 2007.

[37] Xilinx, “Optimizing FPGA power with ISE design tools,” Xcell Journal, Issue 60, pp. 16-19, 2007.

[38] R. Tessier, V. Betz, D. Neto, A. Egier, and T. Gopalsamy, “Power-Efficient RAM mapping algorithms for FPGA embedded memory blocks,” IEEE Trans. of Computer- Aided Design, vol. 26, no. 2, pp. 278-289, Feb 2007.

[39] V. George, H. Zhang, and J. Rabaey, “The design of a low energy FPGA,” Proc. Int. Symp. on Low Power Electronics and Design, pp. 188-193, 1999.

[40] S. Sivaswamy, G. Wang, C. Ababei, K. Bazargan, R. Kastner, and E. Bozargzadeh, “HARP: hard-wired routing pattern FPGAs,” Proc. Int. Symp. on Field-Prog. Gate Arrays, pp. 21- 29, 2005.

[41] E. Kusse and J. Rabaey, “Low-energy embedded FPGA structures,” Proc. Int. Symp. Low Power Electronics and Design, pp. 155-160, 1999.

[42] J.H. Anderson and F.N. Najm, “A novel low-power FPGA routing switch,” Proc. IEEE Custom Integrated Circuits Conf., pp. 719-722, 2004.

[43] Y. Lin, F. Li, and L. He, “Routing track duplication with fine grained power-gating for FPGA interconnect power reduction,” Proc. Asia South Pacific Design Automation Conf., pp. 645-650, 2005.

[44] J. Lamoureux, G.G. Lemieux, and S.J.E. Wilton, “Glitchless: dynamic power minimization in FPGAs through edge alignment and glitch filtering,” (VLSI) SYSTEMS, VOL. 16, NO. 11, NOVEMBER 2008

[45] L. Cheng, D. Chen, and M. Wong. GlitchMap: An FPGA technology mapper for low power considering glitches. In ACM/IEEE DAC, pages 318 –323, 2007.

[46] Q. Dinh, D. Chen, and M. Wong. A routing approach to reduce glitches in low power FPGAs. In ACM ISPD, pages 99–106, 2009.

[47] Warren Shum and Jason H. Anderson Department of Electrical and Computer Engineering, University of Toronto “FPGA Glitch Power Analysis and Reduction”, IEEE 2011

[48]Clock Power Reduction for Virtex-5 FPGAs by Qiang Wang, Xilinx, Inc. Subodh Gupta, Xilinx, Inc. Jason

Anderson, ECE Dept., Univ. of Toronto- FPGA’09, February 22–24, 2009, Monterey, California,

[49] A.Dollas, K.Papademetriou, E. Sotiriades “A case study of Rapid Prototyping of Hardware Systems: the effect of CAD tools capabilities, Design flows and Design Styles” Proceeding of IEEE International Workshop on Rapid System Protyping, pp 180-86 June 2004.

[50] P.Garrault and B. Phiofsky “HDL Coding Practices to Accelerate Design Problems” Xilinx Corporation Tech Rep.,January 2006

[51] I.Kuon and J.Rose “Measuring the gap between FPGAs and ASICs” In FPGA 2006.